計算機アーキテクチャ　95－1
(1992. 8. 19)

分散共有メモリ型マルチプロセッサ「ＡＳＵＲＡ」の階層性とその評価

城 和貴†, 柳原 守†, 田中高士†, David FRASER†, 新田 博之†
齋藤 秀樹‡, 森 眞一郎‡, 富田眞治‡
阿草清滋◇

†：(株) クボタ　‡：京都大学 工学部,　◇：名古屋大学 工学部,

分散共有メモリ型マルチプロセッサ「ＡＳＵＲＡ」の性能評価を確率モデルを利用して行なう。モデル化にあたっては、ＡＳＵＲＡの特徴である階層性に着目し、１サイクル内のキャッシュ、ローカル・メモリ、グローバル・メモリに対する可能なリクエスト数をそれぞれ求め、それらを統合してシステム全体の理論性能値とする。クラスタ内の評価にはマルコフ・チェイン・モデルを、クラスタ間の評価には予想されるリクエストのキューイング数をそれぞれ組み込み、全体として統合されたときの誤差を少なくする手法を提案する。さらにそのモデルを用いてキャッシュのヒット率及びリクエストの分散度合と理論性能との関係を分析する。

Hierarchy Impact and Performance Evaluation of ASURA: A Distributed Shared Memory Multiprocessor

Kazuki JOE†, Mamoru YANAGIHARA†, Takashi TANAKA†,David FRASER†, and Hiroyuki NITTA†
Hideki SAITO‡, Shin-ichiro MORI‡, Shinji TOMITA‡
Kiyoshi AGUSA◇

†: Office of Computer Business, KUBOTA Corporation
ASTEM RI
17 Chudoji, Minami-machi, Shimogyo-ku, Kyoto 600 Japan

‡ : Department of Information Science
Faculty of Engineering, Kyoto University
Yoshida-hon-machi, Sakyo-ku, Kyoto 606-01 Japan
◇ : Department of Electrronic Information
Faculty of Engineering, Kyoto University
Yoshida-hon-machi, Sakyo-ku, Kyoto 606-01 Japan

E-mail: {joe, yanagi, takashi, david, nitta}@kocb.astem.or.jp E-mail: {saito, moris, tomita}@kuis.kyoto-u.ac.jp
E-mail: agusa@nuee.nagoya-u.ac.jp

A new analytic performance model is presented for ASURA, a distributed shared memory multiprocessor. The system bandwidth is analyzed as the total number of requests in a system cycle. Following the fundamental hierarchy of ASURA, the requests are devided into cache, local memory and global memory types. Futhermore, we adopt the adaptive method which increases the accuracy of overall model: 1) Markov Chain Model for the analysis of intra cluster activity. 2) Queueing probability for inter cluster requests. Finally, using this model we analyze the system performance from the viewpoint of cache hit ratio and the distribution of requests.

1 はじめに

プロセッサ・クラスタ方式の並列計算機は将来のスーパーコンピューティングへの有力な一方式として、商用 [26]、研究用 [12][8][9] 共に研究開発が推進されている。我々は既に大規模な並列処理計算機への第一歩として、クラスター・ベースの階層型マルチプロセッサ・システムＡＳＵＲＡを提案している [27]。そこで本稿では、ＡＳＵＲＡの階層性に着目した確率モデルを検討し、性能評価を行なう。

並列計算機の性能評価のためのモデル化は単純なクロスバ結合 [5][11][14] やマルチバス [20][6] に対するものから、クラスター・ベース [16][15][3][2][1] のものまで、さまざまな手法が提案されている。本稿で検討する確率モデルを用いたもの以外にも、マルコフ・モデル [18]、セミ・マルコフ・モデル [19]、ペトリ・ネットワーク・モデル [17]、Ｍ／Ｇ／１モデル [21] 等さまざまな手法が提案され、アーキテクチャの定量的性能評価に利用されている。

階層的な構造を持つクラスター・ベースの並列計算機の性能評価を検討する場合、その階層性を無視してデータのアクセス・パタンを一様と仮定すると、計算される性能は究めて悪くなることが報告されている [16]。そのため、数年前のクラスター・ベースの並列計算機の評価モデルには、「Favorite Memory」の概念が導入された [7][15][3]。これはソフトウェアの観点から見た場合、数値計算等で自然に発生するデータの局所性、さらにその最適化 [13]、これを一般化したクラスタ間のデータ分割配置 [22][10] といった諸問題に何らかの有効な手段が施された結果と見ることが出来よう。そして、近い将来、システムの階層性に特有なデータ・アクセスの分布はパラメータとして与えることが出来るものと期待される。本稿での手法も、ＡＳＵＲＡの階層性に適合すると思われるデータ・アクセスの分布を仮定して、性能評価を検討していく。

本稿ではＡＳＵＲＡの概要を述べた後、モデルの仮定を示し、各階層のモデルを統合する方式でシステム全体のモデルを表し、性能評価を行なう。

2 ＡＳＵＲＡの概要

ＡＳＵＲＡとは株式会社クボタと京都大学とが共同で開発している実験システムである [27][25]。図１にそのアーキテクチャを示す。ＡＳＵＲＡはメモリの階層性に重点を置いたプロセッサ・クラスタ方式の密結合型マルチプロセッサ・システムである。

プロセッシング・エレメント（ＰＥ）はオンチップの１次キャッシュと４ＭＢの２次キャッシュを持つＲ４０００ＭＣである。これらのローカル・キャッシュはスヌープ方式のキャッシュ・コヒーレンス制御によるイリノイ・プロトコル [4] を一部変更したものに従う。

８個のＰＥと２５６ＭＢのローカル・メモリ、ネットワーク・インタフェース（ＮＩＦ）をバスによって結合し、１つのプロセッサ・クラスタ（ＰＣ）を構成する。このローカル・メモリはクラスタ内のＰＥによって共有される。ＮＩＦは１６ＭＢのグローバル・メモリと３２ＭＢのグローバル・キャッシュから構成される。グローバル・メモリはＡＳＵＲＡシステムの分散共有メモリとして全てのＰＥによって共有される。グローバル・キャッシュはクラスタ内のＰＥで共有され、フルマップ・ディレクトリ方式のキャッシュ・コヒーレンス制御によるシナプス・プロトコル [4] に従う。また、グローバル・キャッシュは上位のキャッシュとの間にＭＬＩ特性を持つため [27] １次及び２次キャッシュはグローバル・キャッシュをキャッシング出来る。

複数のＰＣを階層型のネットワークによって結合することによりＡＳＵＲＡシステムが形成される。現段階では、４つのＰＣをレジスタ・インサーション・リング [23] を一部変更した方式でリング結合しクラスタ・グループを形成する。さらに３２クラスタ・グループをクロスバ結合することにより、最大１０２４ＣＰＵを構成することが出来る。

ＡＳＵＲＡの特徴はそれぞれ階層的なキャッシュ、メモリ、ネットワークである。キャッシュは前述したように３階層から成り、メモリはあるＰＥから見てローカル・メモリ、クラスタ内のグローバル・メモリ、クラスタ・グループ内のグローバル・メモリ、それ以外のグローバル・メモリと４階層からなる。また、ネットワークはクラスタ内のバス、クラスタ・グループ内のリング、全体にわたるクロスバと３階層からなる。

3 準備

3.1 ＡＳＵＲＡのモデル化

性能評価のモデル化のために、ＡＳＵＲＡを次のように定義する。Ｐ個のプロセッサ（プライベート・キャッシュ（ＰＣＨ）を持つ）と１つの[1]メモリ・モジュール、ネットワーク・インターフェイス（ＮＩＦ）がバス結合され、プロセッサ・クラスタ（ＰＣ）を形成する。ただし、ＮＩＦはグローバル・メモリ（ＧＭ）とグローバル・キャッシュ（Ｇ

[1] 実際にはインターリーブされているが、バスとメモリのサイクル比が同じと考えた方がモデルが簡単であるためこのような仮定を用いる

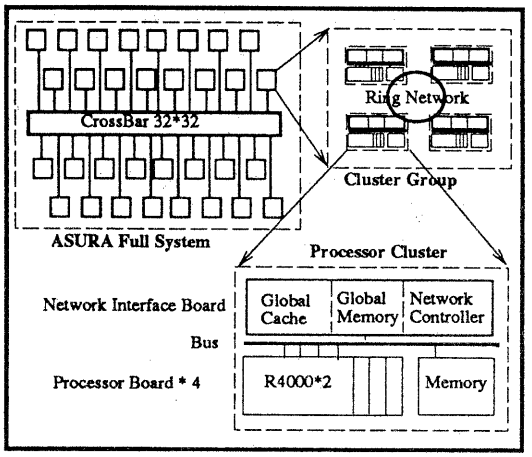

図 1: ＡＳＵＲＡのアーキテクチャ

ＣＨ)、ＰＣ間の通信を制御するコントローラからなる。(ＰＣ間の通信はＰＣ内のバスに影響を与えないことに注意) M個のＰＣはリング結合されてクラスタ・グループ（ＣＧ）を構成し、K個のＣＧがクロスバ結合することによりＡＳＵＲＡ全体が定義される。従って、全体ではＰ×Ｍ×Ｋ個のプロセッサとＰＣＨ、それぞれＭ×Ｋ個のローカル・メモリ・モジュール、グローバル・メモリ・モジュール、ＧＣＨ、Ｍ×Ｋ本のバス、Ｋ個のリング、１個のクロスバによって並列計算機システムを構成することになる。メモリ階層性を示すために、メモリ・ロケーションをローカル・メモリ（ＬＭ)、ＰＣ内グローバル・メモリ（ＰＣＧＭ)、ＣＧ内グローバル・メモリ（ＣＧＧＭ)、リモート・グローバル・メモリ（ＲＧＭ）で区別する。

3.2 モデルの仮定

本稿ではＡＳＵＲＡの階層的なメモリ及びネットワークの性能を評価するために、システム全体のバンド幅を確率的なモデル [20][14][6][7] によって求める。モデルを簡略化するために次のような仮定を導入する。

1. 全てのプロセッサからのリクエストは全て独立である
2. リクエスト先ははある確率分布関数に従う
3. アクセプトされなかったリクエストはキャンセルされる
4. ネットワークの遅延時間はリング以外は無視する
5. ＰＣＨ、ＧＣＨのコヒーレント動作は考慮に入れない

仮定２はデータの局所性を意味している。キャッシュはデータの局所性を利用した機構であり、様々なアーキテクチャに利用できる。これは並列計算機におけるキャッシュでも同じであり、この性質をメモリにまで適用すると、ＬＭ、ＰＣＧＭ、ＣＧＧＭ、ＧＭの参照確率が一様ではなく、参照時間と参照確率の頻度とのトレードオフによって階層的なモデルを形成出来る [24]。例えば、確率密度関数Ｐｒを、$Pr[X_{PCH}] > Pr[X_{GCH/LM}] > Pr[X_{PCGM}] > Pr[X_{CGGM}] > Pr[X_{GM}]$ のように設定出来る。ただし、同一階層においては参照確率は一様であるとする。例えば、ＣＧ内のリクエストが特定のＰＣである確率は１／Ｍである。同様、ＲＧＭへのリクエストが特定のＣＧである確率は１／Ｋである。

仮定３は幾分非現実的である。なぜならば、アクセプトされなかったリクエストは次のサイクルで再発行されるからである。しかしながら、この仮定はモデルを大幅に簡単にし、実際の結果との誤差も小さいと報告されている [5]。本モデルでは、ＰＣ内のみ仮定３の条件を緩和して、マルコフ・チェイン・モデルを用いた手法 [11] で、リクエスト率の誤差修正を図る。

仮定４ではリングの遅延はバスやクロスバのそれと比較して無視できないので、リングの遅延についてはモデルの対象としている。

並列計算機の定量的評価を行なう際には、評価の中心であるバンド幅の定義を明確にしなければならない。これまでに提案されている手法には、

- プロセッサとメモリのサイクルが等しく、ネットワークの遅延は無視するという仮定の上で、１サイクルに対するリクエスト数を持って全体のバンド幅とするもの [16][3]
- プロセッサとメモリのサイクル比を与え、メモリのサイクルごとにビジー状態のメモリ・モジュールの数を算出したものを持って全体のバンド幅とするもの [20]
- プロセッサのスピードを与え、そのスピードとクラスタ数でネットワークの遅延を近似し、適当なプロセッサ利用率を与えて全体の性能を算出するもの [1]

などがある。本モデルでは、ネットワークが複数個あることから、最もクリティカルな部分、すなわちＰＣ内バスのサイクル・タイムを基準にして、プロセッサ及びその他のネットワークのサイクル・タイムの比を与え、基準のバス・サイクルでのシステム全体（プライベート・キャッシュ、ローカル・メモリ、グローバル・キャッシュ、グローバル・メモリ）のリクエスト数を持ってシステム全体のバンド幅と定義する。

3.3 記数法

Ｐ ＰＣ内のプロセッサ数
Ｍ ＣＧ内のＰＣ数

K 全体のＣＧ数
W_{CG} ＣＧ内でのリクエストの平均待ちサイクル数
L ＣＧ内の各ノードでのリクエストの最大キューイング数
BW システム全体のバンド幅
BW_{local} ローカル・メモリに対するバンド幅
BW_{global} グローバル・メモリに対するバンド幅
Ψ プロセッサのリクエスト率
Ψ_{LM} プロセッサのＬＭに対するリクエスト率
Ψ_{GCH} プロセッサのＧＣＨに対するリクエスト率
Ψ_{PCGM} プロセッサのＰＣＧＭに対するリクエスト率
Ψ_{CGGM} プロセッサのＣＧＧＭに対するリクエスト率
Ψ_{RGM} プロセッサのＲＧＭに対するリクエスト率
Ψ_{ring} リングに対するリクエスト率（リングのサイクル基準）
S_{LM} リクエスト先がＬＭである確率
S_{PCGM} リクエスト先がＰＣＧＭである確率
S_{CGCM} リクエスト先がＣＧＣＭである確率
S_{RGM} リクエスト先がＲＧＭである確率
h_{PCH} ＰＣＨに対するヒット率
h_{GCH} ＧＣＨに対するヒット率
α プロセッサとバスのサイクル・タイムの比
β バスとリング・ネットワークのサイクル・タイムの比
P_{CG} リング・ネットワークにＰＣから入力される確率
P_X ＣＧからクロスバ・ネットワークに入力される確率
$q_{CG}(i)$ ＣＧにｉ個のリクエストが発行される確率
$E_{CG}(i)$ ＣＧに発行されたｉ個のリクエストがアクセプトされる確率

4 モデル

4.1 ＰＣのモデル

一般にＰＣＨを持つＰ個のプロセッサが１個のメモリモジュールにバス結合されている場合を考える。プロセッサのサイクルごとのリクエスト率をΨ、キャッシュのヒット率をｈ、プロセッサとバスのサイクルの比をαとした場合、あるプロセッサがバスの１サイクル内にリクエストを出す確率は

$$\alpha(1-h)\Psi \tag{1}$$

よって、少なくとも１つのプロセッサがバスのサイクル内にリクエストを出す確率は

$$1-(1-\alpha(1-h)\Psi)^P \tag{2}$$

仮定より、バスへのリクエストはそのままメモリへの参照と見なされる。また、各プロセッサはバスのサイクル毎に$\alpha h \Psi P$回のリクエストを自分のＰＣＨに発行している。よって、ＰＣのバンド幅は

$$BW_{(0)} = 1-(1-\alpha(1-h)\Psi)^P + \alpha h \Psi P \tag{3}$$

バスへのリクエスト数は$\alpha(1-h)\Psi$Ｐであるから、リクエストがアクセプトされる確率$P_A^{(0)}$は

$$P_A^{(0)} = \frac{BW_{(0)}}{\alpha(1-h)\Psi P} \tag{4}$$

ここで、前章仮定３の条件を緩和してリクエスト率の誤差修正を図る。もともとの仮定ではキャンセルされたリクエストは捨てさられるわけだが、実際は同じリクエストを次のサイクルで再発行することになる。これを、１サイクルの待ち状態の後、即座に別のリクエストを発行すると考えると、プロセッサの状態は、リクエストを発行しているアクティブな状態(A)と、キャンセルされた待ち状態(W)の２状態に分類され、その状態推移はマルコフ・チェインを形成する。図２はその状態推移を表す。a1,a2,a3,a4はそれぞれ、状態(A)から

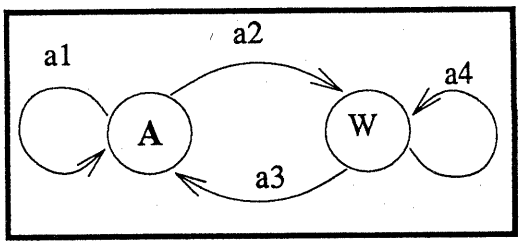

図２: 状態推移を示すマルコフ・グラフ

(A)、(A)から(W)、(W)から(A)、(W)から(W)への推移確率で、以下のように表される。（ただし、$\psi = \alpha(1-h)\Psi$とする。）

$$a1 = (1-\psi) + P_A^{(0)}\psi \tag{5}$$
$$a2 = \psi(1-P_A^{(0)}) \tag{6}$$
$$a3 = P_A^{(0)} \tag{7}$$
$$a4 = 1-P_A^{(0)} \tag{8}$$

a1,a2,a3,a4は推移確率行列を形成するので、以下のようにして確率ベクトルを求めることが出来る。

$$\lim_{m\to\infty}\begin{pmatrix} a1 & a2 \\ a3 & a4 \end{pmatrix}^m = \frac{1}{P_A^{(0)}+\psi(1-P_A^{(0)})}\begin{pmatrix} P_A^{(0)} & \psi(1-P_A^{(0)}) \\ P_A^{(0)} & \psi(1-P_A^{(0)}) \end{pmatrix} \tag{9}$$

よって、状態(A),(W)の確率Q_A, Q_Wは

$$Q_A = \frac{P_A^{(0)}}{\psi+(1-P_A^{(0)})} \tag{10}$$
$$Q_W = 1-Q_A \tag{11}$$

となる。状態(A)の時のリクエスト率はΨQ_A、状態(W)の時のリクエスト率はQ_Wだから、修正されたリクエスト率$\Psi^{(1)}$は

$$\Psi^{(1)} = \Psi^{(0)} Q_A + Q_W \tag{12}$$

これを$P_A^{(0)}$を使って表すと、

$$\Psi^{(1)} = \frac{\{\alpha(1-h)(1-P_A^{(0)}) + P_A^{(0)}\}\Psi}{\psi + P_A^{(0)}(1-\psi)} \tag{13}$$

よって

$$BW^{(1)} = 1 - (1-\Psi^{(1)})^P + \alpha h \Psi^{(1)} P \tag{14}$$

以下同様に計算して、

$$BW_{local} = \lim_{k\to\infty} BW^{(k)} \tag{15}$$

を得る。

4.2 CGのモデル

一般にM個のノードがリング結合されている場合を考える。ノードのサイクルごとのリクエスト率をΨ、ノードとリングのサイクルの比をβとする。まず、リングのサイクルに対してi個のリクエストが発行される確率q(i)は

$$q(i) = \binom{M}{i} \left(\frac{\Psi}{\beta}\right)^i \left(1-\frac{\Psi}{\beta}\right)^{M-i} \tag{16}$$

リングのサイクルにi個のリクエストがなされたとき、それらがアクセプトされる数をE(i)とする。i個のリクエストのリクエスト先の総数はM^i。今、ある特定のノードはリクエストを受けないとすると、リクエストの総数は$(M-1)^i$となるので、特定のノードが少なくとも1つリクエストを受ける確率は

$$1 - \left(\frac{M-1}{M}\right)^i \tag{17}$$

よって、M個のノードに対してi個のリクエストがなされたときにアクセプトされる確率は

$$E(i) = M - M\left(\frac{M-1}{M}\right)^i \tag{18}$$

仮定4で述べたように、リング・ネットワークの遅延時間は無視できないため、リクエストがアクセプトされる時間の近似を行なわなければCGのバンド幅が推定できない。リングがアイドル状態ならリクエストはMサイクルで実行されるとする。CG内の各ノードでのリクエストの最大キューイング数をLとすると、CG内でのリクエストの平均待ちサイクル数は

$$W = \sum_{i=0}^{LM} i \binom{M+i}{i} \left(\frac{\Psi}{\beta}\right)^i \left(1-\frac{\Psi}{\beta}\right)^M \tag{19}$$

よって、CGでのバンド幅は

$$\frac{\sum_{i=0}^{M} E(i)\, q(i)}{M + W_{CG}} \tag{20}$$

式を簡単にすると

$$\frac{M}{M+W}\left\{1 - \left(1 - \frac{\Psi}{M\beta}\right)^M\right\} \tag{21}$$

4.3 クロスバ・ネットワークの分析

K×Kのクロスバ・ネットワークの1入力に、リクエスト率rの入力が与えられたとする。K個の出力のうち特定の1出力にこのリクエストが伝えられる確率はr/Kである。特定の1出力にK個のどの入力のリクエストも伝えられない確率は

$$\left(1 - \frac{r}{K}\right)^K \tag{22}$$

よって、ある特定の出力に対して少なくとも1つの入力が伝えられる確率は

$$1 - \left(1 - \frac{r}{K}\right)^K \tag{23}$$

4.4 全体のモデル

P個のPCH付きプロセッサと1つのローカル・メモリ、NIFをバス結合したものをPC、M個のPCをリング結合したものをCG、K個のCGをクロスバ結合したものをASURAの現在の最大構成とする[2]。ただし、NIFには分散共有メモリの一部であるグローバル・メモリとGCHが含まれている。このグローバル・メモリはプロセッサからのアクセス時間によって、PCGM、CGGM、RGMの3階層に分れる。

システム全体のバンド幅BWは、ローカル・メモリ及びグローバル・キャッシュに対するバンド幅をBW_{local}、グローバル・メモリに対するバンド幅をBW_{global}とすると、

$$BW = M\,K\,BW_{local} + K\,BW_{global} \tag{24}$$

で求められる。

プロセッサのローカル・メモリへのリクエスト率をΨ、PCHのヒット率をh_{PCH}、GCHのヒッ

[2]現在のところP×M×K = 1024

ト率をh_{GCH}、プロセッサとバスのサイクル・タイムの比をαとする。また、リクエスト先がＬＭ、ＰＣＧＭ、ＣＧＧＭ、ＲＧＭである確率をそれぞれ、S_{LM}、S_{PCGM}、S_{CGGM}、S_{RGM}とする。この時、

$$S_{LM}+S_{PCGM}+S_{CGGM}+S_{RGM}=1 \quad (25)$$

が成り立つ。プロセッサからＬＭ、ＧＣＨ、ＰＣＧＭ、ＣＧＧＭ、ＲＧＭにそれぞれ実際に[3]リクエストされた確率、Ψ_{LM}、Ψ_{GCH}、Ψ_{PCGM}、Ψ_{CGGM}、Ψ_{RGM}は以下のように表される。

$$\Psi_{LM}=(1-h_{PCH})S_{LM}\,\alpha\Psi \quad (26)$$

$$\Psi_{GCH}=(1-h_{PCH})h_{GCH} \times(S_{PCGM}+S_{CGGM}+S_{RGM})\,\alpha\Psi \quad (27)$$

$$\Psi_{PCGM}=(1-h_{PCH})(1-h_{GCH})S_{PCGM}\,\alpha\Psi \quad (28)$$

$$\Psi_{CGGM}=(1-h_{PCH})(1-h_{GCH})S_{CGGM}\,\alpha\Psi \quad (29)$$

$$\Psi_{RGM}=(1-h_{PCH})(1-h_{GCH})S_{RGM}\,\alpha\Psi \quad (30)$$

よってローカル・メモリ及びグローバル・キャッシュに対するバスへのリクエスト率は

$$\psi=\Psi_{LM}+\Psi_{GCH}+\Psi_{PCGM} \quad (31)$$

式 (2)(25) を利用して

$$BW^{(0)}_{local}= \{S_{LM}+S_{PCGM}+h_{GCH}(S_{CGGM}+S_{RGM})\} \times\{1-(1-\psi)^P\}+\alpha h_{PCH}\Psi P \quad (32)$$

以下 4.1と同様に計算を行ない、

$$P_A^{(0)}=\frac{BW^{(0)}_{local}}{\alpha\Psi P} \quad (33)$$

$$\Psi^{(1)}=\frac{\psi(1-P_A^{(0)})+\Psi P_A^{(0)}}{P_A^{(0)}+\psi(1-P_A^{(0)})} \quad (34)$$

$$BW^{(1)}=1-(1-\Psi^{(1)})^P+\alpha h_{PCH}\Psi^{(1)}P \quad (35)$$

を繰り返すことにより

$$BW_{local}=\lim_{k\to\infty}BW^{(k)}_{local} \quad (36)$$

を得る。

ＣＧのリング・ネットワークにＰＣから入力される確率をP_{CG}、ＣＧからクロスバ・ネットワークに入力される確率をP_Xとする。Ψ_{CGGM}とΨ_{RGM}は発行されたＰＣの外へのリクエストであるから、リングとバスとのサイクル・タイムの比をβとすると、ＰＣ内の少なくとも１つのプロセッサが自分のＰＣ外にリクエストを出す確率、すなわちP_{CG}は、

$$P_{CG}=1-\left(1-\frac{\Psi_{CGGM}+\Psi_{RGM}}{\beta}\right)^P \quad (37)$$

これがＣＧ内のリクエストである確率は

$$P_{CG}\frac{S_{CGGM}}{S_{CGGM}+S_{RGM}} \quad (38)$$

ＣＧ外のリクエストである確率は

$$P_{CG}\frac{S_{RGM}}{S_{CGGM}+S_{RGM}} \quad (39)$$

よって、ＣＧ内の少なくとも１つのＰＣがＣＧ外にリクエストを出す確率は

$$P_X=1-\left(1-P_{CG}\frac{S_{RGM}}{S_{CGGM}+S_{RGM}}\right)^M \quad (40)$$

P_Xはクロスバ・ネットワークの１入力に与えられる確率で、伝えられるべき出力は自分以外のK-1個の出力のどれかである。式 (23) を利用すると、クロスバ・ネットワークの特定の出力に少なくとも１つのＣＧからの入力が伝わる確率は、

$$1-\left(1-\frac{P_X}{K-1}\right)^{K-1} \quad (41)$$

これは特定のＣＧがクロスバ・ネットワーク経由でリクエストを受ける確率に等しいので、結局リングのリクエスト率Ψ_{ring}は

$$\Psi_{ring}= P_{CG}\frac{S_{RGM}}{S_{CGGM}+S_{RGM}}+1-\left(1-\frac{P_X}{K-1}\right)^{K-1} \quad (42)$$

以上のことより、ＧＣ内にｉ個のリクエストが発行される確率$q_{CG}(i)$、それらがアクセプトされる確率$E_{CG}(i)$、平均待ち時間W_{CG}は式(16)(18)(19)よりそれぞれ

$$q_{CG}(i)=\binom{M}{i}\Psi_{ring}^i(1-\Psi_{ring})^{M-i} \quad (43)$$

$$E_{CG}(i)=M-M\left(\frac{M-1}{M}\right)^i \quad (44)$$

$$W_{CG}=\sum_{i=0}^{LM} i\binom{M+i}{i}(\Psi_{ring})^i(1-\Psi_{ring})^M \quad (45)$$

であるから、グローバル・メモリに関するバンド幅は式(20) より

$$\frac{M}{M+W_{CG}}\left\{1-\left(1-\frac{\Psi_{ring}}{M}\right)^M\right\} \quad (46)$$

[3] ＰＣＨにヒットしないという意味

よって、全体のバンド幅は式(24)(15)(46) より

$$BW = K\,M\,BW_{local} + K\,BW_{global} \qquad (47)$$

によって求められる。

5 評価

これまでに述べてきたモデルを用いて、ASURAの性能評価を行なう。性能評価の対象となるパラメータは、h_{PCH} と S_{LM} である。ただし、$h_{GCH} = 0.8h_{PCH}$、$S_{CGGM} = (1 - S_{LM})/4$、$S_{PCGM} = S_{CGGM}/4$、$S_{RGM}$=1-$S_{PCGM} - S_{CGGM} - S_{RGM}$ という共通の設定を行なっている。

h_{PCH} を固定する場合 (0.9 に固定) S_{LM} の値を、S_{LM} を固定する場合 (0.6 に固定) h_{PCH} の値を、それぞれ 0 から 1 まで 0.01 間隔に変えて評価を行なった。また、固定したパラメータは、$\alpha = 1.0, \beta = 1.0, L = 5$ である。

図 3はASURA全体のバンド幅を示している。h_{PCH} を固定した場合は S_{LM} の値にあまり関係なく高いバンド幅を示している。また S_{LM} を固定した場合、ほぼ h_{PCH} の値に従った性能向上を示している。

図 4はPCにおけるPCH、LM、GCH、PCGMのバンド幅を示している。図 3とほとんど同じような性能向上を示していることから明らかに、ここの性能がシステム全体の性能を決定していると言えよう。

図 5はCGにおけるCGGM、RGMのバンド幅を示している。パラメータの値によって、かなり複雑な結果を示しているが、全体に占めるバンド幅の量が極めて少ないため、システム全体の性能にはあまり影響を与えないと言えよう。

6 結論

分散共有メモリ型マルチプロセッサ「ASURA」の基本性能を、その階層性に着目した確率モデルで近似し、データの分散度合、及びキャッシュのヒット率が全体の性能にどのように影響を与えるかを確認した。その結果、全体の性能は主にキャッシュのヒット率に影響を受け、高ヒット率では理論性能に近い性能が可能であることを確認した。

本稿ではキャッシュのコヒーレント動作に関する定量化がなされなかったが、本モデルを拡張してキャッシュのコヒーレント動作を組み込んだ性能評価のモデルを開発すること、及び全体の正確なシミュレーションを行なうことが今後の研究課題である。

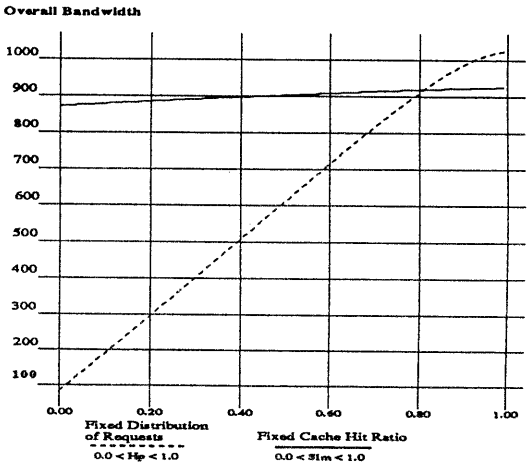

図 3: ASURA全体の性能

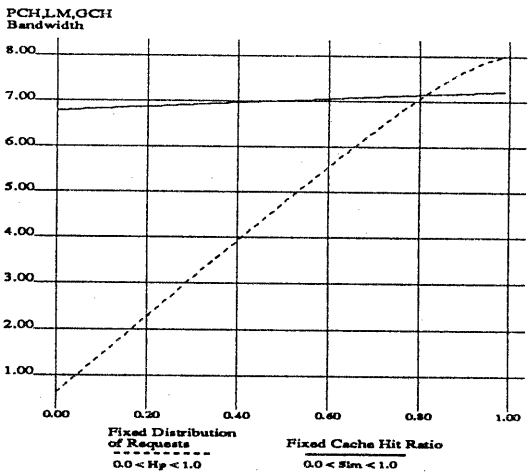

図 4: キャッシュ、ローカル・メモリに関する性能

謝辞

日頃ご討論頂く (株) クボタ コンピュータ事業推進室 山口宗之部長, 及び同R&Dグループ の諸氏に感謝致します.

参考文献

[1] Santosh G. Abraham and Edward S. Davidson. A communication model for optimizing hierarchical multiprocessor systems. In *Proceedings of the 1986 International Conference on Parallel Processing*, pages 467–474, 1986.

[2] Dharma P. Agrawal and Imadeldin O. Mahgoub. Performance analysis of cluster-based supersystems. In *Proceedings of the International Symposium on Computer Architecture*, pages 593–602, 1985.

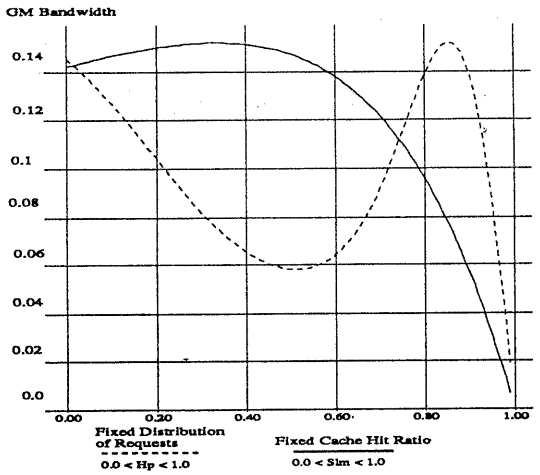

図 5: グローバル・メモリに関する性能

[3] Dharma P. Agrawal and Imadeldin O. Mahgoub. Analysis of a class of cluster-based multiprocessor systems. *Information Science International Journal*, 43:85–105, 1987.

[4] James Archibald and Jean-Loup Baer. Cache coherence protocols: Evaluation using a multiprocessor simulation model. *ACM Transactions on Computer Systems*, 4(4):273–298, 1986.

[5] D. P. Bhandarkar. Analysis of memory interference in multiprocessors. *IEEE Transactions on Computers*, 24(9):897–908, 1975.

[6] Laxmi N. Bhuyan. A combinatorial analysis of multibus multiprocessors. In *Proceedings of the 1984 International Conference on Parallel Processing*, pages 225–227, 1984.

[7] Laxmi N. Bhuyan. An analysis of processor-memory interconnection networks. *IEEE Transactions on Computers*, 34(3):279–283, 1985.

[8] Daniel Lenoski et al. The directory-based cache coherence protocol forr the dash multiprocessor. In *Proceedings of the International Symposium on Computer Architecture*, pages 148–159, 1990.

[9] David R. Cheriton et al. Paradigm: A highly scalable shared-memory multicomputer architecture. *IEEE Computer*, pages 33–46, 1991.

[10] Manish Gupta and Prithviraj Banerjee. Demonstration of automatic data partitioning techniques for parallelizing compilers on multicomputers. *IEEE Transactions on Parallel and Distributed Systems*, 3(2):179–193, 1992.

[11] Kai Hwang and Faye A. Briggs. *COMPUTER ARCHITECTURE AND PARALLEL PROCESSING*. McGraw-Hill, 1985.

[12] David J. Kuck, Edward S. Davidson, and Duncan H. Lawrie Ahmed H. Sameh. Parallel supercomputing today and the cedar approach. 電子情報通信学会論文誌, J71-D(8):1361–1374, 1988.

[13] Monica S. Lam, Edward E. Rothberg, and Michael E. Wolf. The cache performance and optimizations of blocked algorithms. In *Proceedings of the Fourth International Conference on ASPLOS*, pages 63–74, 1991.

[14] Tomas Lang, Mateo Valero, and Ignacio Alegre. Bandwidth of crossbar and multiple-bus connections for multiprocessors. *IEEE Transactions on Computers*, 31(12):1227–1234, 1982.

[15] Imadeldin O. Mahgoub and Dharma P. Agrawal. Impact of cluster network failure on the performance of cluster-based supersystems. In *Proceedings of the 1986 International Conference on Parallel Processing*, pages 743–749, 1986.

[16] Imadeldin O. Mahgoub and A. K. Elmagarmid. Performance analysis of a generalized class of m-level hierarchical multiprocessor systems. *IEEE Transactions on Parallel and Distributed Systems*, 3(2):129–138, 1992.

[17] Marco Ajmone Marsan, Gianfranco Balbo, Gianni Conte, and Francesco Gregoretti. Modeling bus contention and memory interference in a multiprocesor system. *IEEE Transactions on Computers*, 32(1):60–72, 1983.

[18] Marco Ajmone Marsan and Mario Gerla. Markov models for multiple bus multiprocessor systems. *IEEE Transactions on Computers*, 31(3):239–248, 1982.

[19] T. N. Mudge and H. B. Al-Sadoun. A semi-markov model for the performance of multiple-bus systems. *IEEE Transactions on Computers*, 34(10):934–942, 1985.

[20] T. N. Mudge, J. P. Hayes, and D. C. Winsor. Multiple bus architectures. *IEEE Computer Magagine*, 20(6):42–48, 1987.

[21] Steven L. Scott, James R. Goodman, and Mary K. Vernon. Performance of the sci ring. In *Proceedings of the International Symposium on Computer Architecture*, pages 403–414, 1992.

[22] Weijia Shang and Jose A. B. Fortes. Independent partitioning of algorithms with uniform dependencies. *IEEE Transactions on Computers*, 41(2):190–206, 1992.

[23] Andrew S. Tanenbaum. *COMPUTER NETWORKS*. Prentice Hall, 1981.

[24] Terry A. Welch. Memory hierarchy configuration analysis. *IEEE Transactions on Computers*, 27(5):408–413, 1978.

[25] 齋藤 秀樹, 森 眞一郎, 城 和貴, David Fraser, 田中 高士, and 富田 眞治. イベント対応型キャッシュ・コヒーレンス制御方式とそのバリア同期への応用. Technical Report 92-ARC-95-2, 情報処理学会計算機アーキテクチャ研究会, 1992.

[26] 日本アライアントコンピュータ. *The CAMPUS/800 Supercomputer*. 技術資料, 1991.

[27] 森 眞一郎, 齋藤 秀樹, 五島 正裕, 富田 眞治, 田中 高士, David Fraser, 城 和貴, and 新田 博之. 分散共有メモリ型マルチプロセッサ「阿修羅」の概要. Technical Report 92-ARC-94-6, 情報処理学会, 1992.